

## Heartbeat+MySQL+DRBD 构建高可用 MySQL

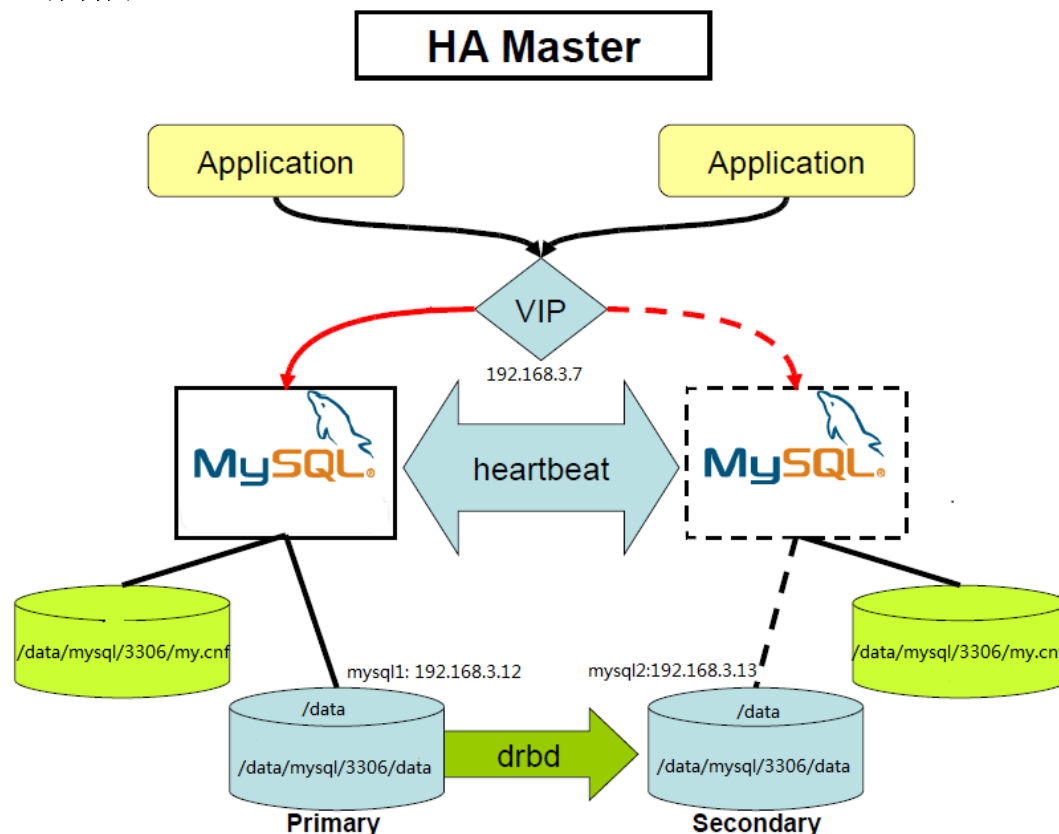
文件状态:	官方网址:	http://www.linuxtone.org
[ ] 草稿	当前版本:	V1.0
[ ✓ ] 正式发布	作者:	NetSeek
[ ] 正在修改	完成日期:	2010年11月20日

### 一、DRBD 原理及架构

#### 1、原理简述

DRBD 是一种块设备,可以被用于高可用(HA)之中.它类似于一个网络 RAID-1 功能.当你将数据写入本地 文件系统时,数据还将会被发送到网络中另一台主机上.以相同的形式记录在一个文件系统中.本地(主节点)与远程主机(备节点)的数据可以保证实时同步.当本地系统出现故障时,远程主机上还会保留有一份相同的数据,可以继续使用.

#### 2、架构图



mysql1: 192.168.3.12

mysql2: 192.168.3.13

VIP: 192.168.3.7

DRBD 资源目录: /data

**优点:** 安全性、稳定性高,出现故障系统将自动切换,从而保证服务的连续性,相对

mysql+heartbeat+存储来说对大优点是节约了存储这个硬件设备。

**缺点：**mysql 主机宕机，切换到备机，备机接管服务，待主机修复完配置 failback 机制的话会发生脑裂情况，需要手工执行命令进行主机接管。

## 二、Heartbeat+DRBD+MySQL 安装

### 1、定义主机名

```
# vi /etc/hosts
```

```
192.168.3.12 mysql2  
192.168.3.13 mysql3
```

### 2、利用 yum 安装 drbd、heartbeat （基于 CentOS x86\_64 平台）

```
yum install -y drbd83 kmod-drbd83  
yum install -y heartbeat heartbeat-ldirectord heartbeat-pils heartbeat-stonith
```

### 3、配置 DRBD

Master Node:

```
# vi /etc/drbd.conf
```

```
# please have a a look at the example configuration file in  
# /usr/share/doc/drbd83/drbd.conf  
#  
global { usage-count yes; }  
common { syncer { rate 100M; } }  
resource r0 {  
    protocol C;  
    startup {  
  
    }  
    disk {  
        on-io-error detach;  
        #size 1G;  
    }  
    net {  
    }  
    on mysql12 {  
        device /dev/drbd0;  
        disk /dev/sda5;  
        address 192.168.3.12:7898;  
        meta-disk internal;
```

```
}
on mysql13 {
    device    /dev/drbd0;
    disk      /dev/sda5;
    address   192.168.3.13:7898;
    meta-disk internal;
}
}
```

### Slave Node:

```
# vi /etc/drbd.conf
```

```
#
# please have a a look at the example configuration file in
# /usr/share/doc/drbd83/drbd.conf
#
global { usage-count yes; }
common { syncer { rate 100M; } }
resource r0 {
    protocol C;
    startup {

    }
    disk {
        on-io-error detach;
        #size 1G;
    }
    net {
    }
    on mysql12 {
        device    /dev/drbd0;
        disk      /dev/sda5;
        address   192.168.3.12:7898;
        meta-disk internal;
    }
    on mysql13 {
        device    /dev/drbd0;
        disk      /dev/sda5;
        address   192.168.3.13:7898;
        meta-disk internal;
    }
}
}
```

在两台主机上执行:

```
# mkfs.ext3 /dev/sda5
# dd if=/dev/zero of=/dev/sda5 bs=1M count=1;sync
```

在两台主机上同时执行：

```
# /etc/init.d/drbd restart
```

查看是状态：

```
# watch -n 1 cat /proc/drbd
```

在主节点主机上执行：

设置当前节点为主节点并进行格式化：

```
# drbdadm -- --overwrite-data-of-peer primary all
# mkfs.ext3 /dev/drbd0
# mount /dev/drbd0 /data
```

继续查看状态：

```
# watch -n 1 cat /proc/drbd
```

#### 4、节点切换

1) 主节点切换：切换主从节点前，要注意 umount

主从切换： drbdadm primary all;drbdadm secondary all

2) 从节点切换：

```
drbdadm primary all
```

```
再 mount /dev/drbd0 /data
```

即可以切换过数据过来

#### 5、配置 heartbeat ldirectord

1) 配置 authkeys

```
# vi /etc/ha.d/authkeys
```

```
auth 3
#1 crc
#2 sha1 HI!
3 md5 Hellomysql
```

```
# chmod 600 vi /etc/ha.d/authkeys
```

2) 配置 ha.cf

```
# vi /etc/ha.d/ha.cf
```

```
debugfile /var/log/ha-debug
logfile /var/log/ha-log
```

```
logfacility    local0
keepalive 2
deadtime 20
warntime 10
initdead 60
udpport 693
ucast eth1 192.168.3.13 #slave 节点配置 ucast eth1 192.168.3.12
ping_group group1 192.168.3.12 192.168.3.13
auto_failback off
node    mysql12
node    mysql13
#respawn hacluster /usr/lib/heartbeat/ipfail
#apiauth ipfail gid=haclient uid=hacluster
hopfudge 1
```

### 3) 配置 haresources

```
# vi /etc/ha.d/haresources
```

```
mysql12          IPaddr::192.168.3.7/24/eth1:0          drbddisk::r0
Filesystem::/dev/drbd0::/data::ext3 mysql
```

4) Slave 节点配置基本类似以上.

### 5) 安装迁移 MySQL

编译安装 MySQL 只将数据库目录存放在 DRBD 目录, 我们要镜像的是 MySQL 数据库目录

### 6) 设置自动启动服务

```
# chkconfig mysqld off
# chkconfig --add heartbeat
# chkconfig heartbeat on
```

## 三、常见问题处理

### 设置一个节点为主节点

通过 `cat /prod/drbd` 查看是否开始复制, 以及两节点的状态。此时开始同步两个节点的磁盘, 需要一定时间, 在同步完成前, 请不要重启, 否则会重新同步。

同步完成后, 如果两个节点都是 Secondary/Secondary, 并且已经 UpToDate 了, 可以挑一个节点设置为主节点:

```
# drbdadm primary all
```

### DRDB 的一些常规操作:

切换主从节点前, 要注意 umount

```
主从切换: drbdadm primary all, drbdadm secondary all
节点间连接: drbdadm connect|disconnect all
```

### DRBD 脑裂后的处理:

脑裂后, 两个节点间数据不同步, 主从关系失效, 需要按下面的步骤修复:

a. 在从节点如下操作:

```
# drbdadm secondary r0
# drbdadm -- --discard-my-data connect r0
```

b. 在主节点上, 通过 `watch -n 1 cat /proc/drbd` 查看状态, 如果不是 `WfConnection` 状态, 需要再手动连接:

```
# drbdadm connect r0
```

### 1). DRBD 协议(A B C)

协议说明

A 数据一旦写入磁盘并发送到网络中就认为完成了写入操作.

B 收到接收确认就认为完成了写入操作.

C 收到写入确认就认为完成了写入操作.

### 2). drbd 设备的三个进程

每个 drbd 设备会有三个进程:

`drbd0_worker` 是 `drbd0` 的主要进程,

`drbd0_asender` 是 `primary` 上 `drbd0` 的数据发送进程,

`drbd0_receiver` 是 `secondary` 上 `drbd0` 的数据接收进程

### 3). 几点注意的地方:

1. `mount drbd` 设备以前必须把设备切换到 `primary` 状态。
2. 两个节点中, 同一时刻只能有一台处于 `primary` 状态, 另一台处于 `secondary` 状态。
3. 处于 `secondary` 状态的服务器上不能加载 `drbd` 设备。
4. 主备服务器同步的两个分区大小最好相同, 这样不至于浪费磁盘空间, 因为 `drbd` 磁盘镜像相当于网络 `raid 1`

### 4). 使用 DRBD 方案的时候建议用使用 INNODB 存储引擎

据说 `MyISAM` 在 `DRBD` 上容易造成数据丢失, 或表文件损坏, 所以此模式下最好用 `InnoDB`, 打开 `Binlog`, 同时, `innodb_flush_log_at_trx_commit=1` 可以保证数据完整, 但会严重影响性能. 设置为 2 的话性能约有 10 倍提升, 不过故障 时可能会丢失 1 秒的数据。

## 四、参考文档

1. <http://www.alidba.net/index.php/archives/65>
2. <http://www.eit.name/blog/read.php?472>
3. <http://blog.thinklet.net/joeyue/tag/drbd/> 【资料管理脚本】

【期待你的加入】LinuxTone 注册邀请码:

<http://bbs.linuxtone.org/home.php?mod=invite&u=1&c=81fa8b4dea5eb4b0>

